

Mengenal Jenis Backup pada SQL Server 2005

Rangga Praduwiratna

ziglaret@yahoo.co.nz

<http://geeks.netindonesia.net/blogs/ziglaret>

Lisensi Dokumen:

Copyright © 2003-2007 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

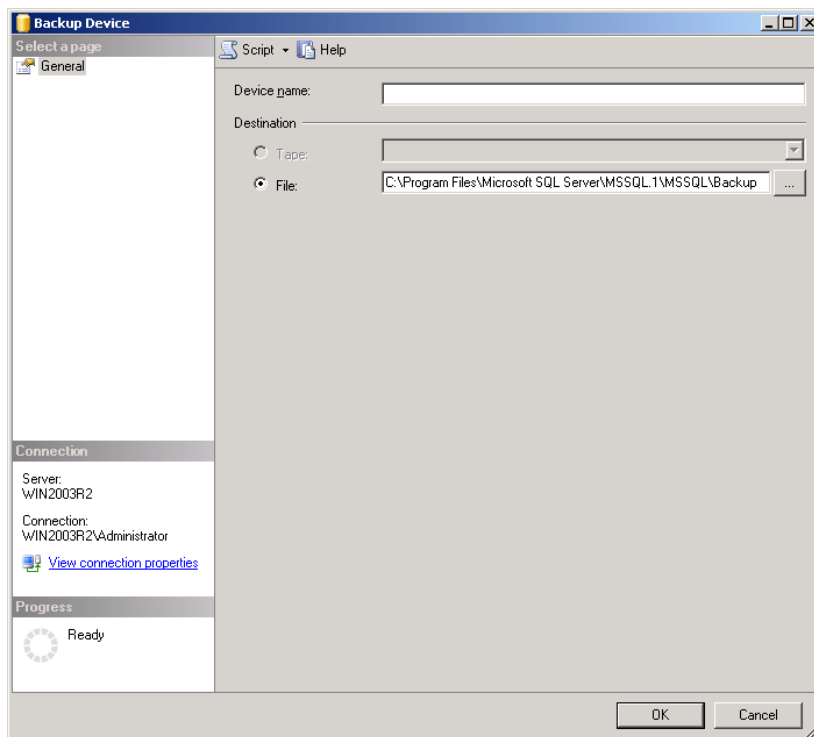
Pendahuluan

Menentukan metoda backup yang tepat untuk database Anda sangat berguna dalam hal administrasi database. Salah satu alasan penting untuk membuat backup database adalah untuk mengurangi resiko hilangnya data yang diakibatkan karena kegagalan atau kerusakan hardware komputer server Anda. Selain itu, dengan membuat backup, Anda juga dapat mengurangi resiko hilangnya data yang diakibatkan oleh kesalahan user atau bencana alam yang dapat terjadi sewaktu-waktu.

Membuat Backup Device pada SQL Server

Backup device adalah device logik yang berfungsi sebagai media backup data dari database Anda. Anda harus mendefinisikan lokasi backup device ini sebelum Anda dapat melakukan backup. Anda dapat membuat backup device dengan cara sebagai berikut :

1. Buka SQL Server Management Studio
2. Expand Server Objects pada Object Exploer SSMS, dan klik kanan pada Backup Devices, pilih New Backup Device.
3. Pada field Device Name, isikan nama backup device Anda, dan pada field File Destination, definisikan lokasi penyimpanan file backup Anda.



Gambar 1. Membuat Backup Device melalui SSMS

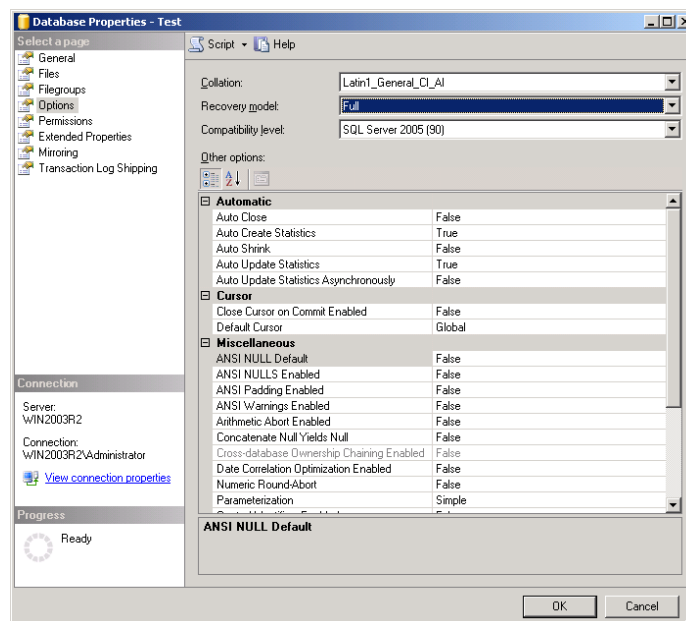
Jenis Backup pada SQL Server

Ada beberapa jenis backup yang tersedia pada SQL Server, namun pada artikel kali ini, saya hanya akan membahas 4 jenis backup : Full Backup, Differential Backup, Incremental Backup, dan Transaction Log Backup.

Full Backup. Jenis backup ini akan membackup semua extent database Anda. Saya pernah sedikit membahas mengenai konsep pages dan extent yang digunakan SQL Server dalam menyimpan data pada harddisk. Extent merupakan kumpulan dari pages berukuran 8 Kb yang digunakan untuk menyimpan data dari database Anda secara logik. Setiap Anda memasukkan data baru pada database, data Anda akan dialokasikan pada extent yang berukuran 64 Kb.

Extent inilah yang dibackup oleh SQL Server agar dapat direstore kembali saat Anda membutuhkannya. Perlu diingat, bahwa sebelum Anda menggunakan metoda backup ini, Anda perlu mensetting recovery model database Anda ke mode Full. Anda dapat mengubah recovery model database Anda dengan cara sebagai berikut :

1. Buka atau expand database Anda pada object explorer di SSMS (SQL Server Management Studio)
2. Klik kanan pada database Anda, dan pilih Properties. Lihat Options Page pada database Anda tersebut.
3. Anda dapat mengubah recovery model database Anda melalui Recovery model drop-down list. Disana terdapat 3 jenis recovery model, yaitu Full, Bulk-Logged, dan Simple.



Gambar 2. Mengatur recovery model database Anda

Anda juga dapat mengubah recovery model database Anda dengan menggunakan query :

```
ALTER DATABASE [database_name]  
SET RECOVERY FULL | BULK_LOGGED | SIMPLE
```

Tentu Anda akan bertanya, seandainya dilakukan full-backup pada bussiness day atau business hour, dimana terdapat kemungkinan database diakses oleh user pada saat sedang dilakukan proses backup, apa yang akan terjadi? Jawabannya adalah database Anda tidak akan dapat diakses oleh user selama dilakukan proses backup. Mengapa? Hal ini dilakukan agar tidak terjadi ketidak konsistenan data saat dilakukan proses backup.

Metoda full backup merupakan metoda yang sangat dianjurkan untuk diimplementasikan pada database produksi. Hal ini dikarenakan metoda ini memiliki konfigurasi pengaturan yang paling baik.

Anda dapat membackup database Anda dengan cara sebagai berikut :

```
BACKUP DATABASE [database_name] TO DISK = '<directory>\<filename>' WITH INIT
```

Parameter INIT digunakan untuk mendefinisikan agar SQL Server melakukan overwrite data yang mungkin telah ada sebelumnya, sedangkan TO DISK mendefinisikan lokasi penyimpanan data backup dari database Anda. Anda perlu membuat direktori yang telah dispesifikasikan pada query sebelum mengeksekusi query ini.

Differential Backup. Jenis backup ini akan membackup semua extent database terakhir Anda setelah Full Backup terakhir dilakukan. Tujuan utama dari metoda backup ini adalah untuk mengurangi jumlah transaction log backup yang harus direstore. Perlu diingat, bahwa metoda ini dilakukan jika Anda telah melakukan full backup pada database Anda.

```
BACKUP DATABASE [database_name] TO DISK = '<directory>\<filename>' WITH DIFFERENTIAL
```

Incremental Backup. Jenis backup ini akan membackup semua extent database terakhir Anda setelah incremental atau differential backup terakhir dilakukan. Metoda ini berbeda dengan metoda differential backup.

Tabel 1. Ilustrasi Metoda Differential Backup

| | Hari | | | |
|---------------------------------|--------|--------|--------|--------|
| | Minggu | Senin | Selasa | Rabu |
| Size database | 100 Mb | 150 Mb | 200 Mb | 250 Mb |
| Jumlah data yang harus dibackup | 100 Mb | 50 Mb | 100 Mb | 150 Mb |

Tabel 2. Ilustrasi Metoda Incremental Backup

| | Hari | | | |
|---------------------------------|--------|--------|--------|--------|
| | Minggu | Senin | Selasa | Rabu |
| Size database | 100 Mb | 150 Mb | 200 Mb | 250 Mb |
| Jumlah data yang harus dibackup | 100 Mb | 50 Mb | 50 Mb | 50 Mb |

Metoda incremental backup membutuhkan semua file backup incremental agar database dapat direstore secara lengkap. Maka dari itu, jika salah satu file backup incremental hilang, maka Anda tidak dapat merestore data pada database secara lengkap. Metoda ini kurang baik diterapkan pada database produksi, karena memiliki resiko yang cukup tinggi jika terjadi *disaster*.

SQL Server mengenali perbedaan data yang terdapat pada database setelah dilakukan full backup dengan cara melihat extents map pada database Anda. Extents database Anda akan direpresentasikan dengan bit mulai dari 0 hingga 1 setiap kali dibuat. Saat Anda melakukan full backup, maka bit-bit extents tersebut akan direset menjadi 0. Dengan cara ini, maka SQL Server dapat mengenali perubahan yang terjadi pada database Anda.

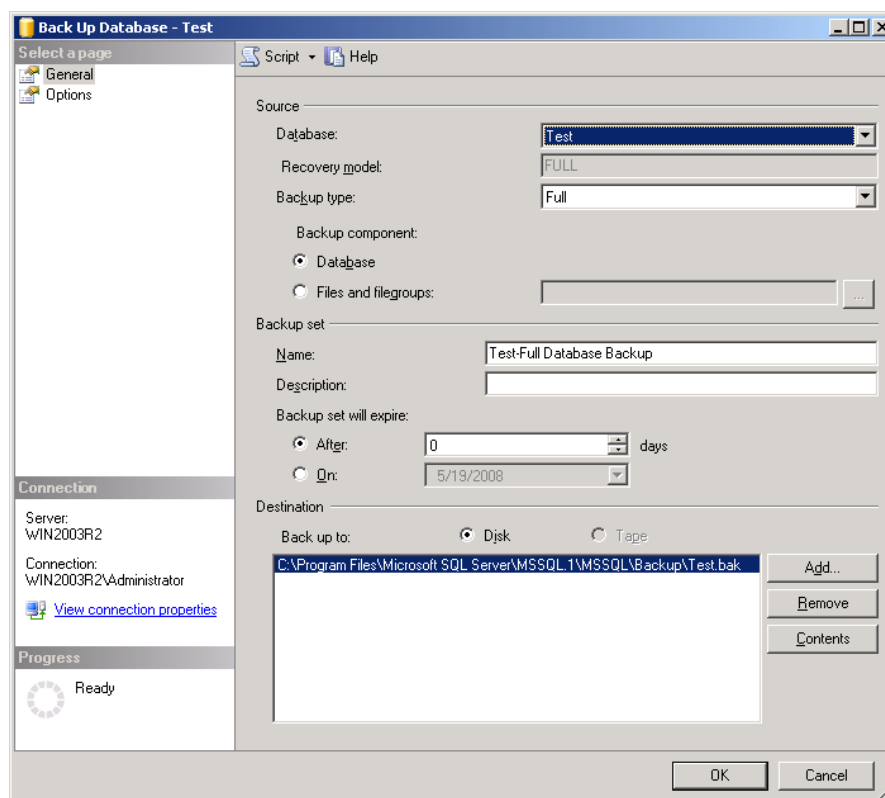
Transaction Log Backup. Metoda backup ini dilakukan untuk membackup file *actives log* database Anda, yang dapat sangat berguna untuk melakukan *tracing*. Untuk dapat menggunakan metoda ini, Anda harus mengubah recovery model database Anda ke mode Full, seperti halnya metoda full backup. Metoda ini juga baru bisa dilakukan jika Anda telah melakukan full backup pada database Anda.

Anda dapat melakukan transaction log backup dengan cara sebagai berikut :

```
BACKUP LOG [database_name] TO DISK = '<directory>\<filename>' WITH INIT
```

Dalam membuat backup dari database Anda, Anda juga dapat menggunakan GUI SSMS (SQL Server Management Studio) Anda. Berikut cara membuat backup dengan menggunakan SSMS :

1. Buka database Anda pada SQL Server Management Studio
2. Klik kanan pada database Anda, dan pilih Task > Backup
3. Pilih database Anda dan jenis backup yang akan dilakukan, serta spesifikasikan juga backup device yang telah Anda buat sebelumnya. Anda dapat melihat pada Options Page untuk mengatur parameter INIT yang terdapat pada query yang telah saya bahas sebelumnya. Klik OK jika Anda telah selesai mengatur konfigurasi backup.



Gambar 3. Membuat backup dengan menggunakan SSMS

Penutup

Menentukan jenis backup yang akan dilakukan merupakan hal yang penting diketahui. Hal ini dapat mengurangi resiko hilangnya data pada database yang diakibatkan oleh kerusakan hardware komputer, bencana alam, kesalahan user, virus, dan lain sebagainya. Untuk selanjutnya, Anda dapat mendelegasikan tugas backup ini pada operator dan membuat scheduled task untuk dijalankan secara otomatis oleh SQL Server. Mengenai hal ini, akan saya bahas pada artikel selanjutnya.

Referensi

- Jorden, Joseph. 2007. SQL Server 2005 DBA Street Smarts. Indiana. Wiley Publishing
- Solid Quality Learning. 2006. SQL Server 2005 Implementation and Maintenance. USA. Microsoft Press
- SQL Server 2005 Book Online (BOL)

Biografi Penulis



Rangga Praduwiratna. Rangga Praduwiratna merupakan mahasiswa jurusan Teknik Informatika Universitas Kristen Maranatha dan Teknik Arsitektur Institut Teknologi Bandung. Ia aktif menulis beberapa artikel mengenai teknologi Microsoft di INDC (Indonesia .NET Developer Community) dan merupakan MCTS (Microsoft Certified Technology Specialist) untuk teknologi SQL Server 2005 – Implementation and Maintenance.

Ia juga sempat menjadi asisten dosen Pemrograman Dasar C di Universitas Kristen Maranatha, dan beberapa kali menjadi pengajar AutoCAD untuk mahasiswa Arsitektur ITB yang diadakan oleh Ikatan Mahasiswa Arsitektur-Gunadharma ITB. Ia memiliki ketertarikan pada teknologi .NET, bahasa pemrograman C#, SQL Server 2005, serta beberapa software desain grafis.